

Addressing Connection Loss in a Mobile, Wireless Environment

Emily Cocanower, Math and CS Department, Lake Forest College, IL 60045 (cocanee@lfc.edu)
Victor Replogle, CS Department, Taylor University, Upland, IN 46989 (vic_replogle@tayloru.edu)
Carl Singer(*), CS Department, DePauw University, Greencastle, IN 46135 (singer@depauw.edu)
Stephen Weaver, CS Department, Millersville University, PA 17551 (sjweaver@cs.millersville.edu)

Handheld computers have become pervasive tools in the past few years. Their portability allows them to be carried with the user anywhere. With the addition of a wireless Internet connection, the user is capable of communicating with other handhelds, traditional computers, and other networked devices. This opens the door for exchanging messages or sharing data among handhelds, other mobile computers, and servers, wherein each device may create and manipulate shared data, and synchronize it across all involved devices. Because handhelds are mobile, movement between access points may result in lost connection. If a user is in the process of performing a data transfer, this creates problems for both the user and the application. By the time the user realizes the connection has been lost, it may not be clear which actions were successfully performed and which were not. Thus, there is a need for connection loss management that ensures the user will be notified and program execution will resume where it left off when a connection is reestablished.

We developed two applications to help explore connection loss in two situations: mobile to server and mobile to mobile communication. The server is assumed to have a stable connection. It does not initiate contact, but only responds to requests. Mobile devices are prone to connection loss depending on signal strength and movement. Mobile devices initiate requests of the server and communication with each other. From these two situations, we developed a generalized connection loss strategy we call Simple State Protocol (SSP).

PocketQuiz uses the first connection type, mobile to server. In *PocketQuiz*, a user downloads a quiz from the server and later uploads the answers to the server. If connection loss occurs during download or upload, the handheld and server will need to remember the connection state at the time of the loss in order to recover. We address this using an implementation of SSP.

PocketChat is a peer to peer chat application. It uses a server to provide IP addresses of the active users so that handhelds can connect directly to one another. If a user changes access points or has intermittent network contact, the connection between handhelds may be lost, which can result in lost messages. When the user regains network connection, however, *PocketChat* will try to reconnect. *PocketChat* uses another implementation of SSP to “fail gracefully” during connection loss and to reconnect without losing messages.

In both situations, Simple State Protocol solves the connection loss problem. It is a general process that can be implemented in various ways to manage data during connection loss. SSP involves keeping track of states on both sides of the connection so that in recovering from a connection loss, the program can tell where it is and resume from that point.

Acknowledgement: This work was supported by National Science Foundation Grant Number EIA-0242293 and by DePauw University